Adaptive Optics (AO)

(and, en passant, also intro to image formation and atmospheric turbulence)

Marcel Carbillet
[marcel.carbillet@oca.eu]

lagrange.oca.eu/carbillet/enseignement/M2-optiQ/

(Credit: Markus Feldt, Max Planck Institut für Astronomie — Heidelberg, 2025)

—> https://www2.mpia-hd.mpg.de/homes/feldt/post/02_adaptive_optics/

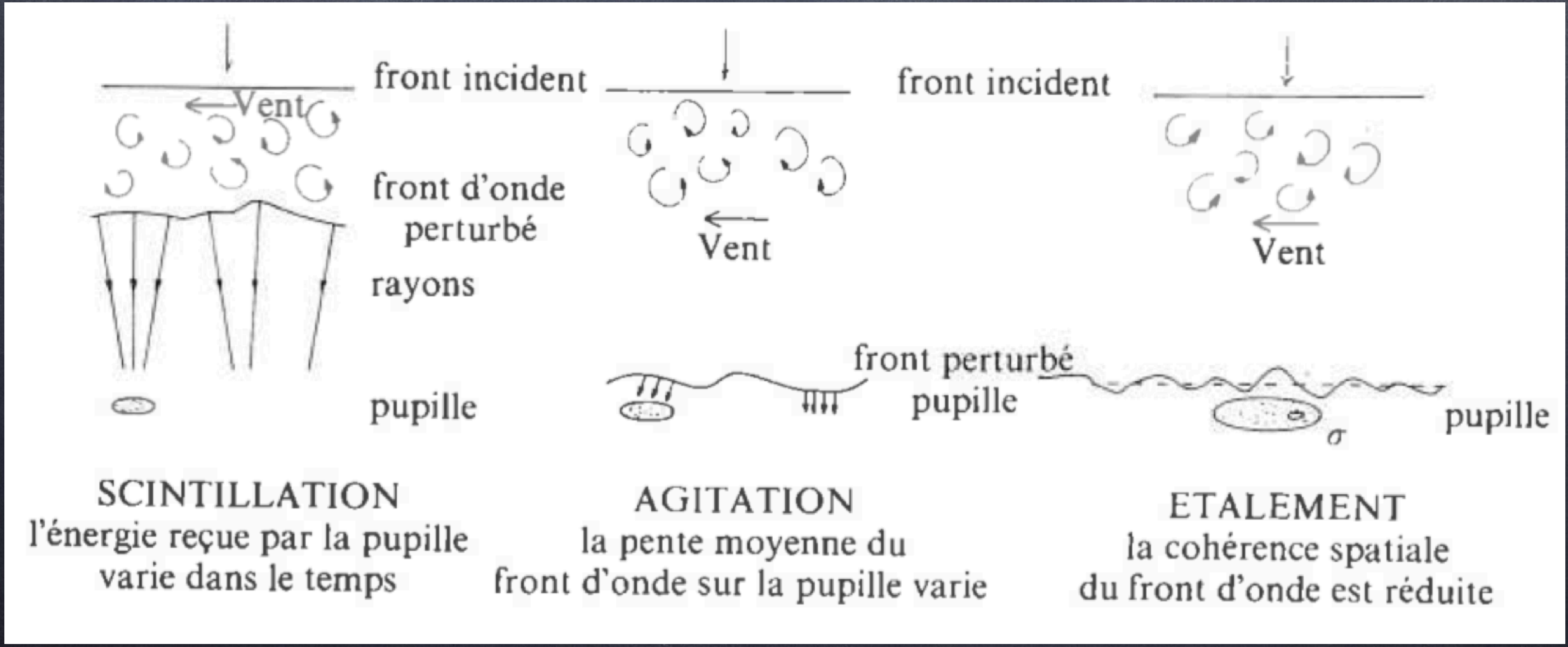
Menu

- High-angular resolution imaging in astronomy
- Atmospheric turbulence
- Numerical modelling of perturbed wavefronts
- Formation of resulting images (+detection noises)
- AO error budget
- Post-AO point-spread function morphology
- Wavefront sensors
- Deformable mirrors
- Reconstruction and control of the command
- Numerical modelling of a complete AO system

Images & turbulence — 01

The image formed through turbulent atmosphere (optically speaking) is degraded:

- Scintillation (due to intensity fluctuation in the pupil).
- Agitation (due to angle-of-arrival variation).
- Spreading (due to a loss of spatial coherence).



(Illustration from Pierre Léna, Astrophysique — Méthodes physiques de l'observation, CNRS Éd. (2me éd.), p.177)

Images & turbulence — 02

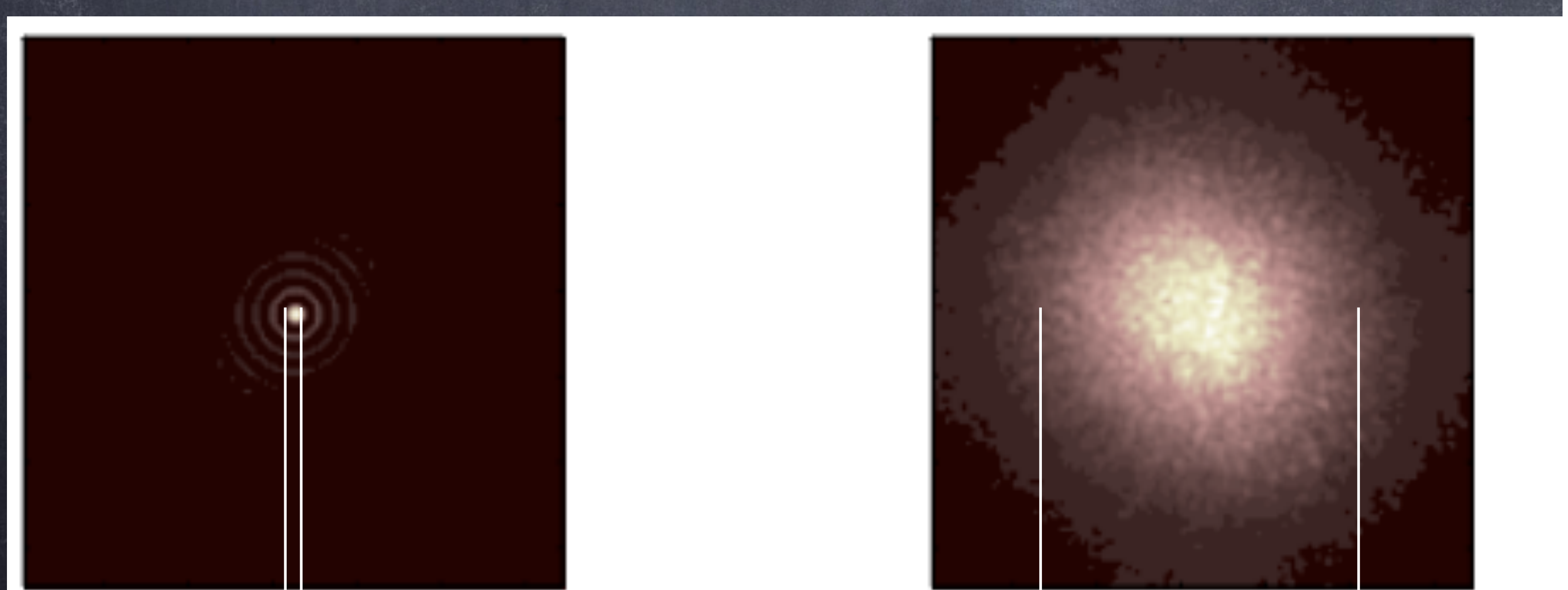
The object-image relation between the intensity *I(α)* in the image plane (i.e. the focal plane of the telescope) and the brightness *O(α)* of the object (in the sky) is a relation of convolution implying the point-spread function (PSF) *S(α)* of the whole ensemble telescope+atmosphere, with *α* the angular coordinates in the focal plane:

$$I(\vec{\alpha}) = O(\vec{\alpha}) * S(\vec{\alpha})$$

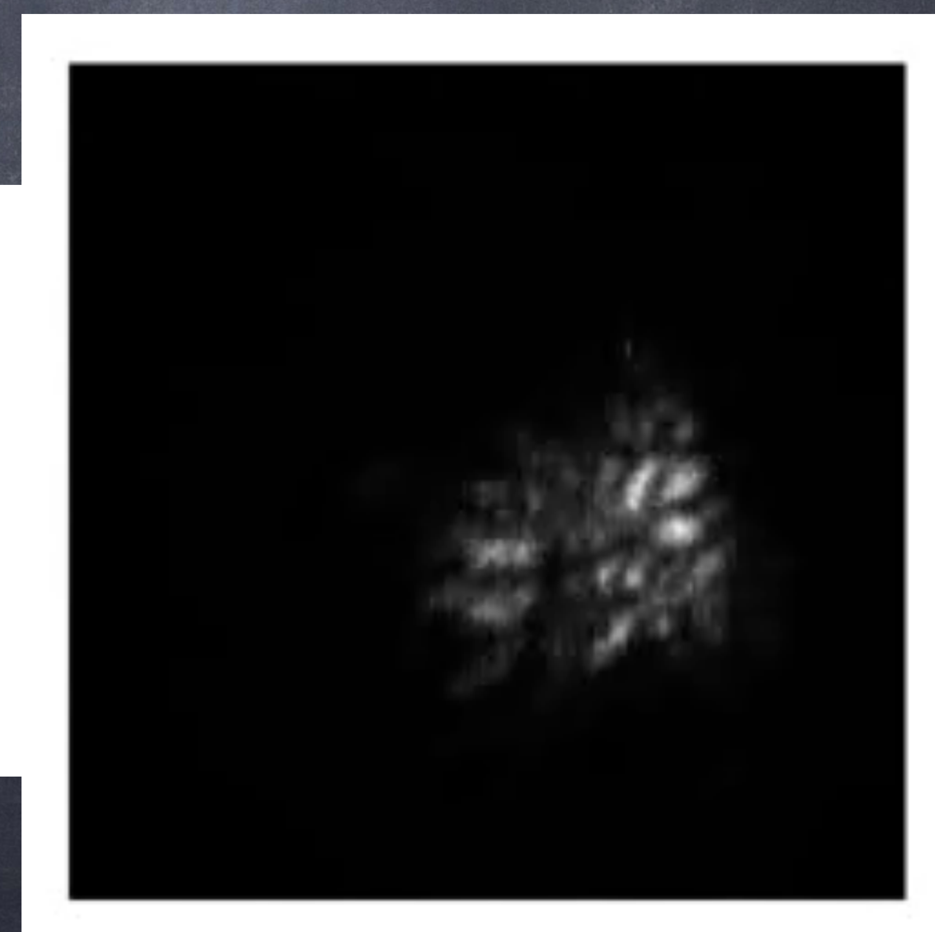
Images & turbulence — 03

$$I(\vec{\alpha}) = O(\vec{\alpha}) * S(\vec{\alpha})$$

This relation is valid notably at the condition that the system is invariant by translation (everything happens within the isoplanatic domain)...



λ/D λ/r_0

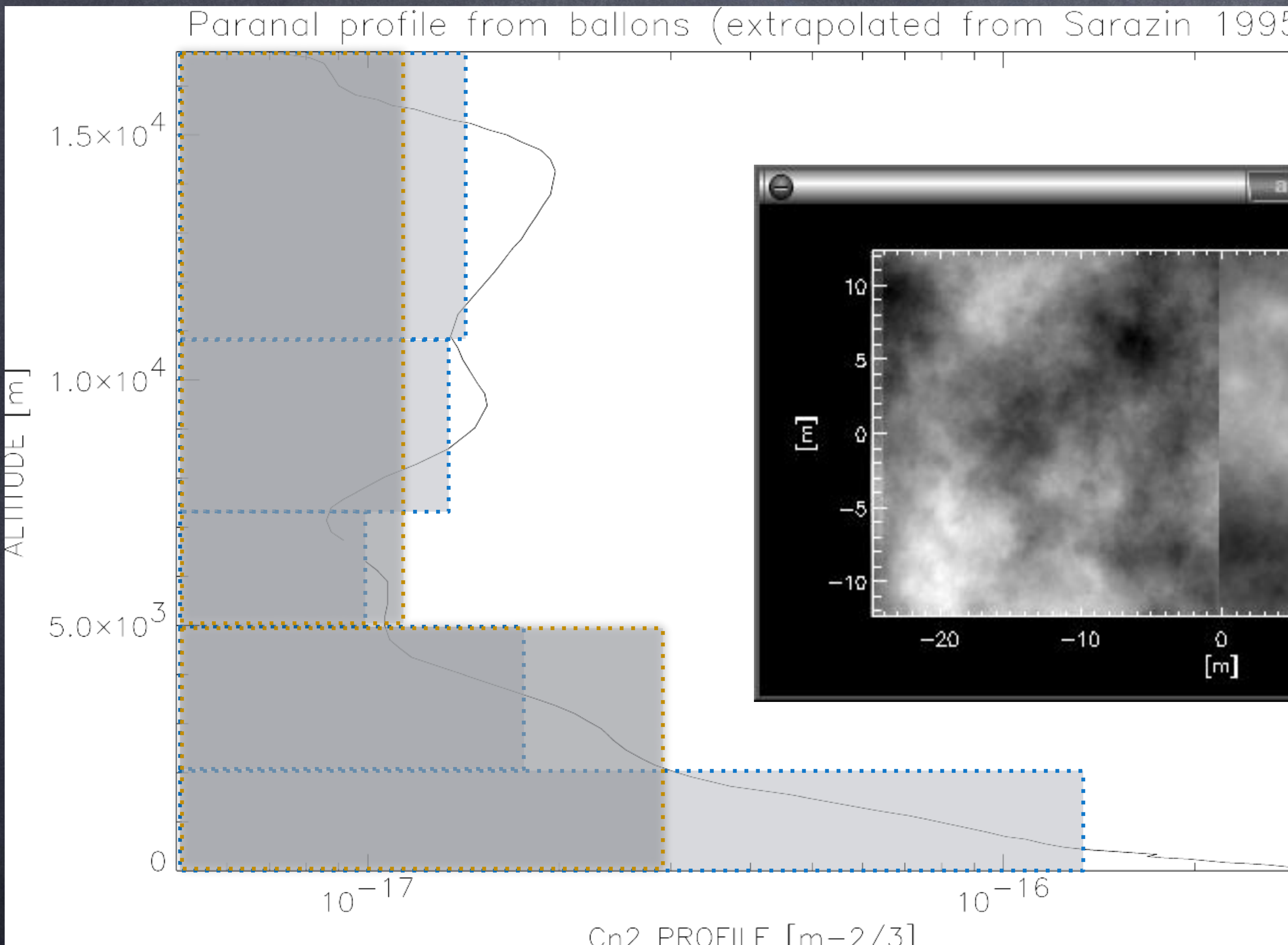


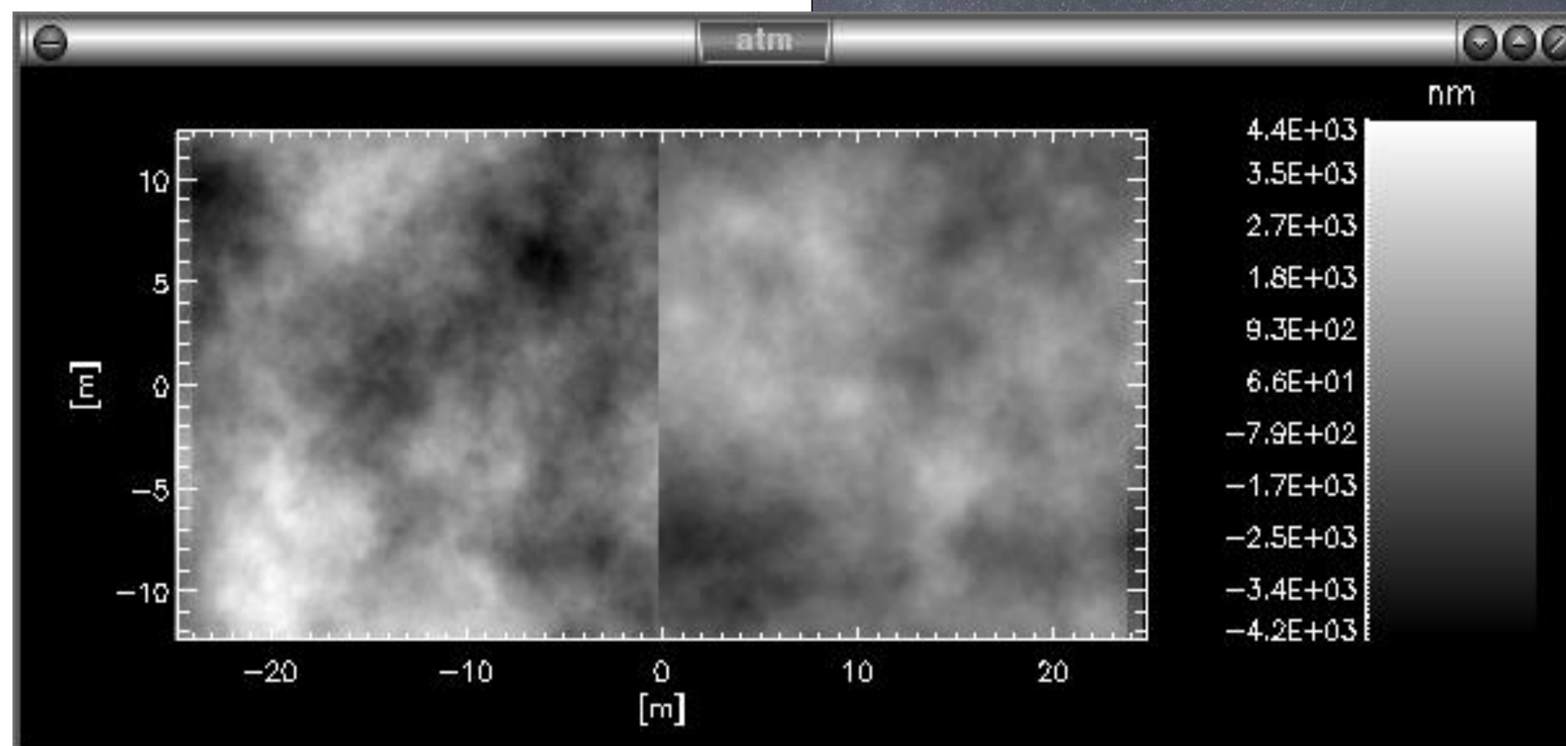
Images & turbulence — 04

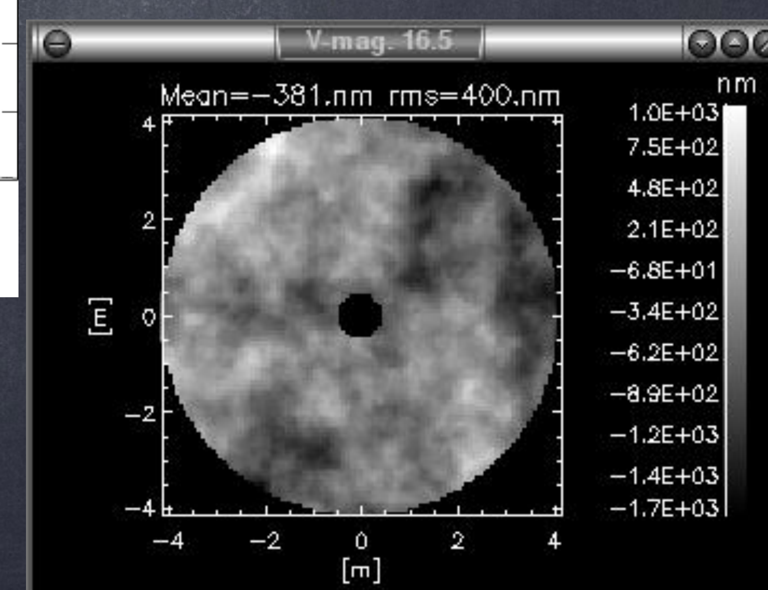
Some orders of magnitude concerning the turbulent atmosphere:

	λ = 500 nm	λ = 2.2 μm
Fried parameter (r_0)	→ 10 cm	60 cm
velocity of the turbulent layers (v)	→ 10 m/s	id.
=> image FWHM ($\epsilon \approx \lambda / r_0$)	→ 1"	~1"
=> evolution time ($\tau_0 \propto r_0 / v$)	→ 3 ms	18 ms

Images & turbulence — 05







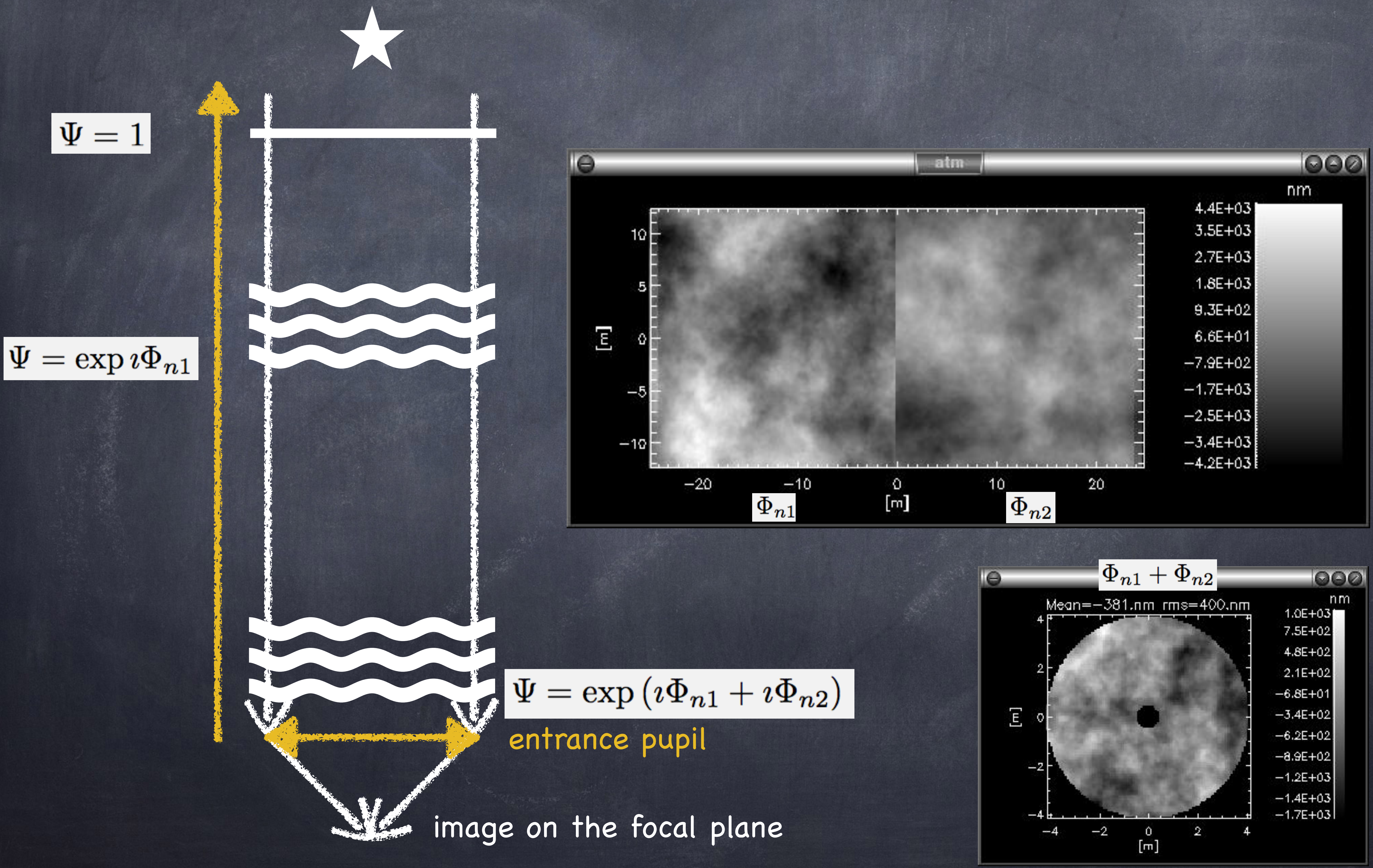
Profile of the refractive index structure constant $C_n^2(h)$

= measure of the intensity of optical turbulence in function of the altitude h

= *constante de structure $C_n^2(h)$ des fluctuations d'indice de réfraction de l'air*

(F. Roddier, 1981)

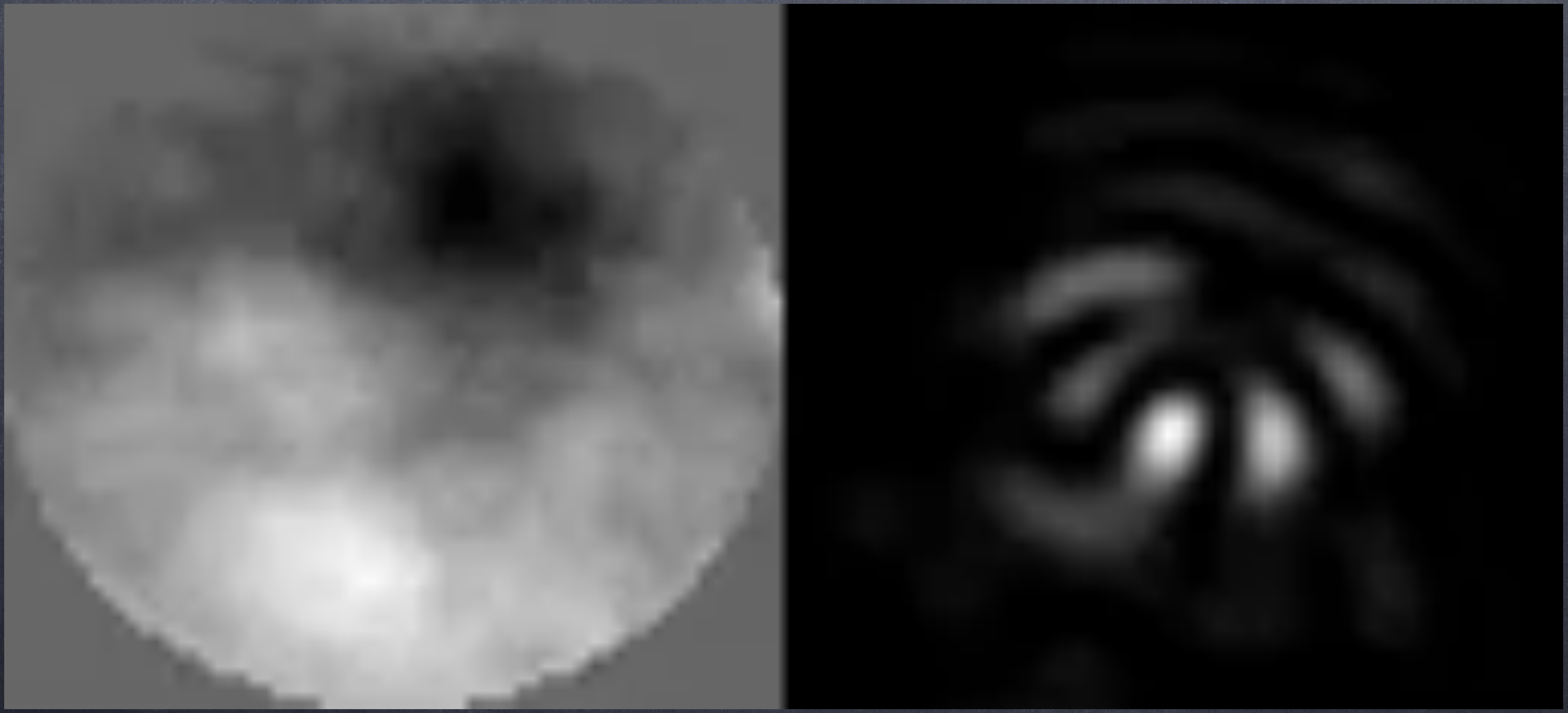
Images & turbulence — 06



Images & turbulence — 07

entrance pupil

image on the focal plane



Images & turbulence — 08

The wavefront is, modulo $\lambda/2\pi$, proportional to the phase $\Phi(r)$ of the wave $\Psi(r)$ which has went through the turbulent atmosphere before reaching the telescope:

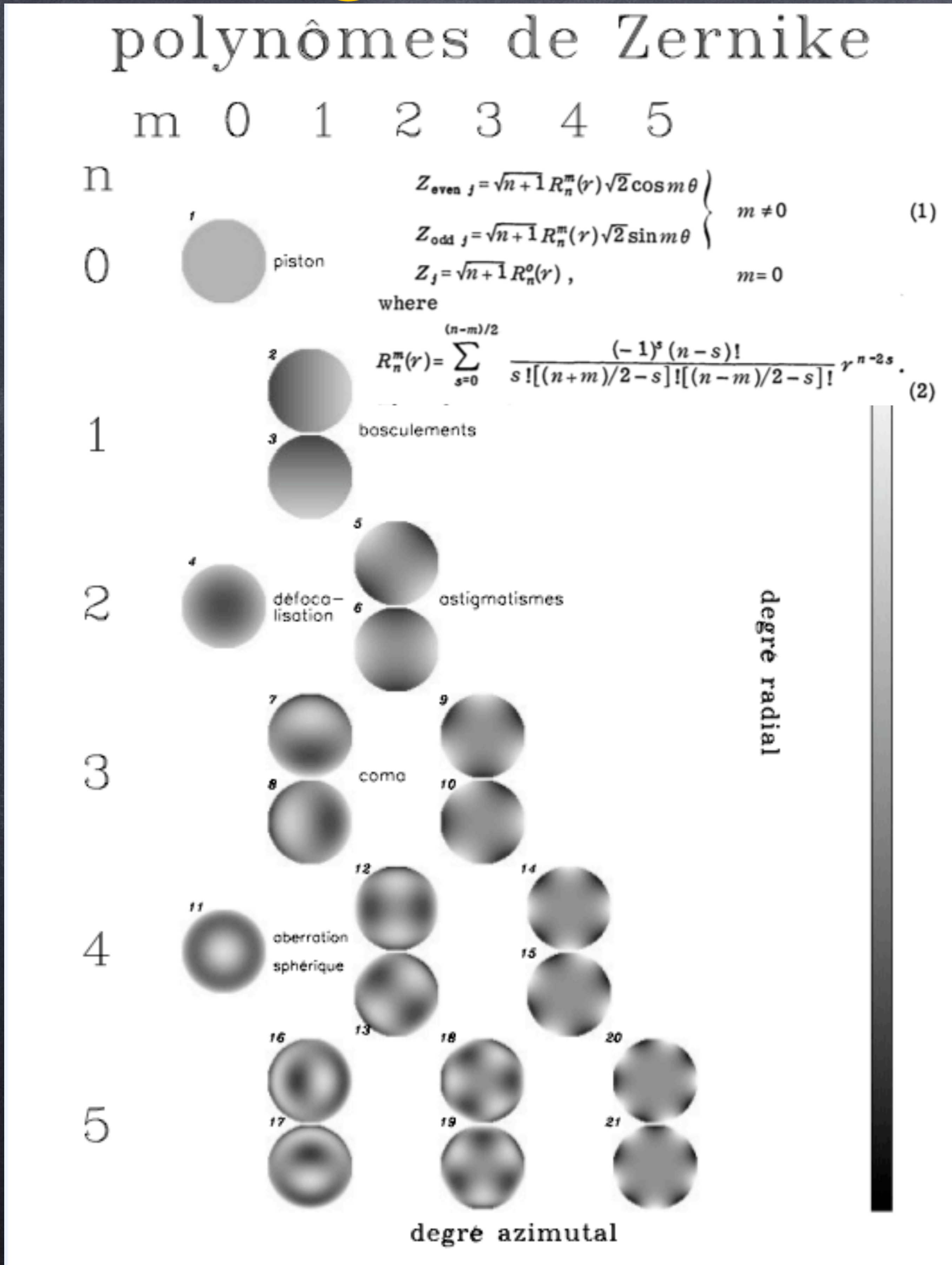
$$\Psi(\vec{r}) = A(\vec{r})\exp\{\imath\Phi(\vec{r})\}$$

Note that this phase can be decomposed following a base of polynomials, for example Zernike ones:

$$\Phi(\vec{r}) = \sum_i a_i\, Z_i(\vec{r})$$

Images & turbulence — 09

entrance pupil



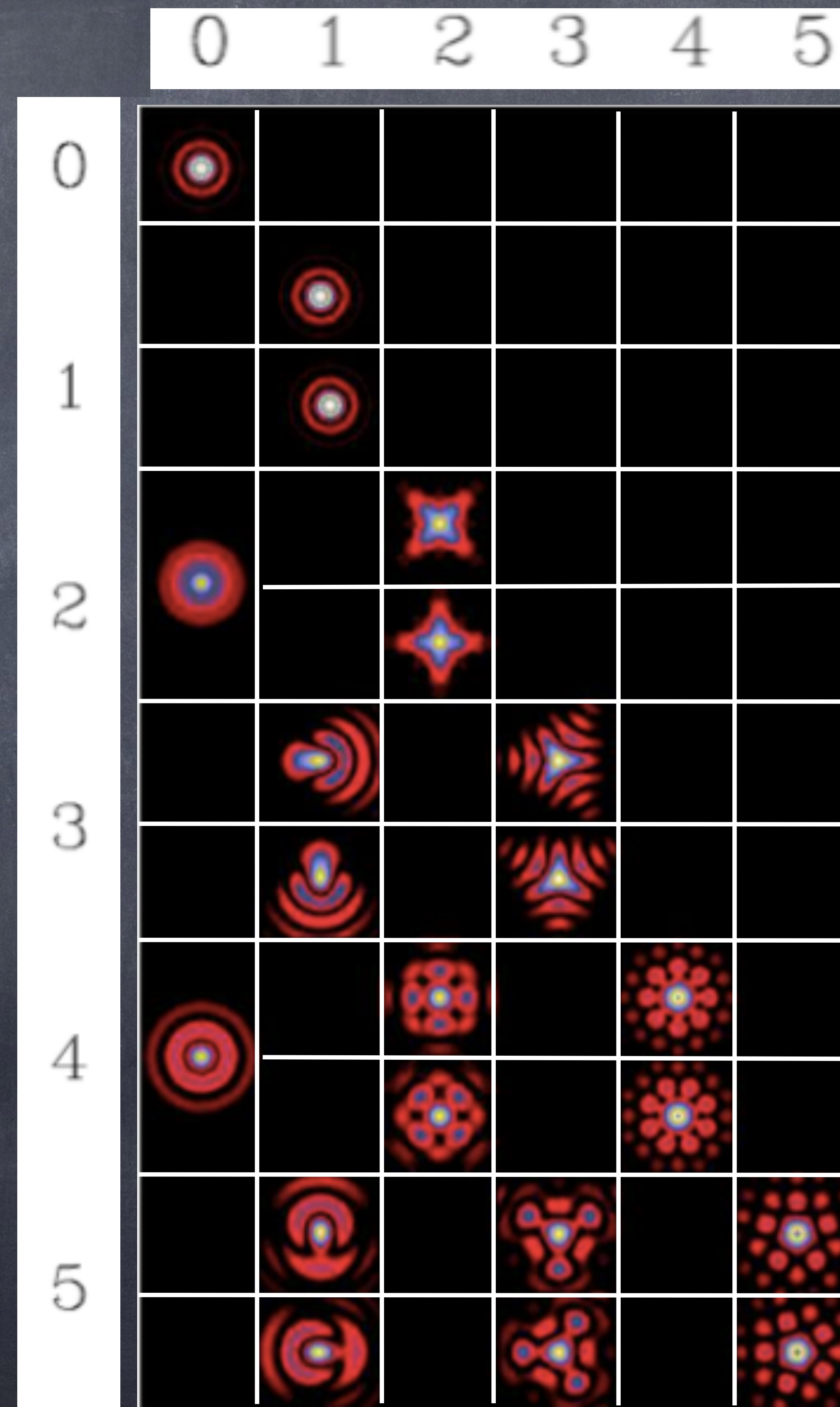


image on the focal plane

Images & turbulence — 10

turbulence intensity [$m^{-2/3}$]

$$r_0 = 0.185\ \lambda^{\frac{6}{5}}\ \cos(\gamma)^{\frac{3}{5}}\ \left[\int_0^\infty C_n^2(z)\ dz\right]^{-\frac{3}{5}}$$

(D.L.Fried 1955, F.Roddier 1981)

=> r0 in band H knowing that r_0@500nm = 10cm ?

$$\tau_0 = 0.36\,\frac{r_0}{\bar{v}}$$

$$\epsilon_0 = 0.98\,\frac{\lambda}{r_0}$$

$$\theta_0 = 0.314\ \frac{r_0}{\bar{h}}$$

$$\bar{v} = \left(\frac{\int C_n^2(h) v(h)^{\frac{5}{3}} dh}{\int C_n^2(h) dh}\right)^{\frac{3}{5}}$$

$$N_s \simeq 0.34\left(\frac{D}{r_0}\right)^2$$

$$\bar{h} = \left(\frac{\int C_n^2(h) h^{\frac{5}{3}} dh}{\int C_n^2(h) dh}\right)^{\frac{3}{5}}$$

Number of speckles for r_0=10cm and D=1m ?

Images & turbulence — 11

r_0 in band H knowing r_0 at 500nm ?...

$$r_0 = 0.185\ \lambda^{\frac{6}{5}}\ \cos(\gamma)^{\frac{3}{5}}\ \left[\int_0^\infty C_n^2(z)\ dz\right]^{-\frac{3}{5}}$$

$$r_0^{\mathrm{H}=1.65\,\mu\mathrm{m}} = r_0^{500\,\mathrm{nm}} \left(\frac{1.65}{0.5}\right)^{\frac{6}{5}} \simeq 0.42$$

Number of speckles for r_0=10cm and D=1m ?...

$$N_S^{500\,\mathrm{nm}} \simeq 0.34\ \left(\frac{1.0}{0.1}\right)^2 \simeq 34$$

$$N_S^{\mathrm{H}} \simeq 0.34\ \left(\frac{1.0}{0.42}\right)^2 \simeq 2$$

Images & turbulence — 12

$$\Phi_\varphi(\vec{\nu}) = 0.0228\ r_0^{-\frac{5}{3}} \left(\nu^2 + \frac{1}{\mathcal{L}_0^2}\right)^{-\frac{11}{6}}$$

Power Spectral Density (PSD) of the phase, function of the spatial frequency

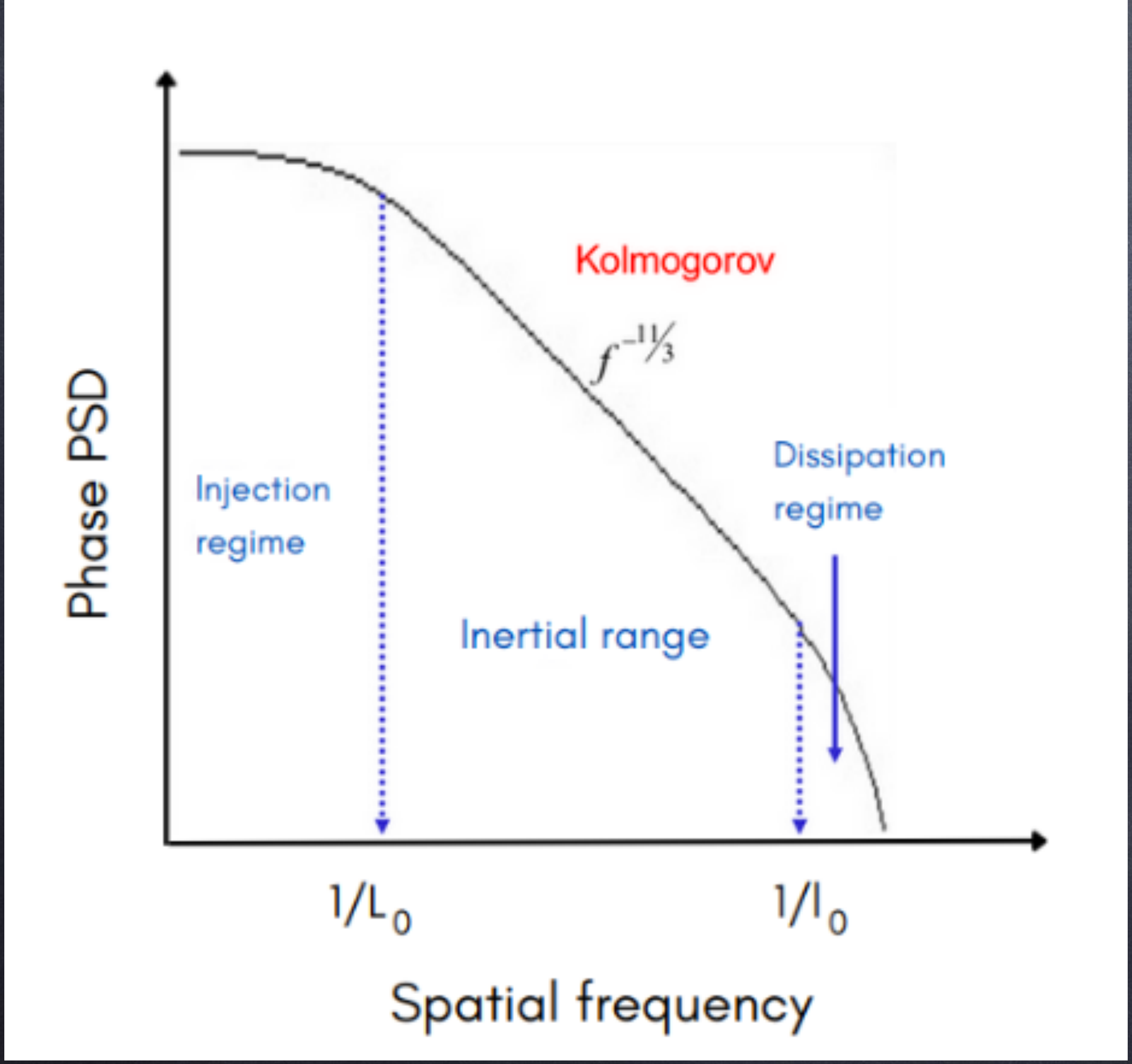
Kolmogorov/von Kármán model

- Kolmogorov : outerscale of turbulence $\mathcal{L}_0$ is infinite.
- One can refine the model by considering also ℓ_0.
- ∃ other models with a finite $\mathcal{L}_0$ (and a non-zero ℓ_0).

Energy cascade: wind shear => turbulent energy injected into the system via a large eddy ($\mathcal{L}_0$) which splits into smaller and smaller eddies (ℓ_0), and is finally viscously dissipated. Interval [ℓ_0, $\mathcal{L}_0$] = **inertial range**.

Images & turbulence — 13

$$\Phi_\varphi(\vec{\nu}) = 0.0228\ r_0^{-\frac{5}{3}} \left(\nu^2 + \frac{1}{\mathcal{L}_0^2}\right)^{-\frac{11}{6}}$$

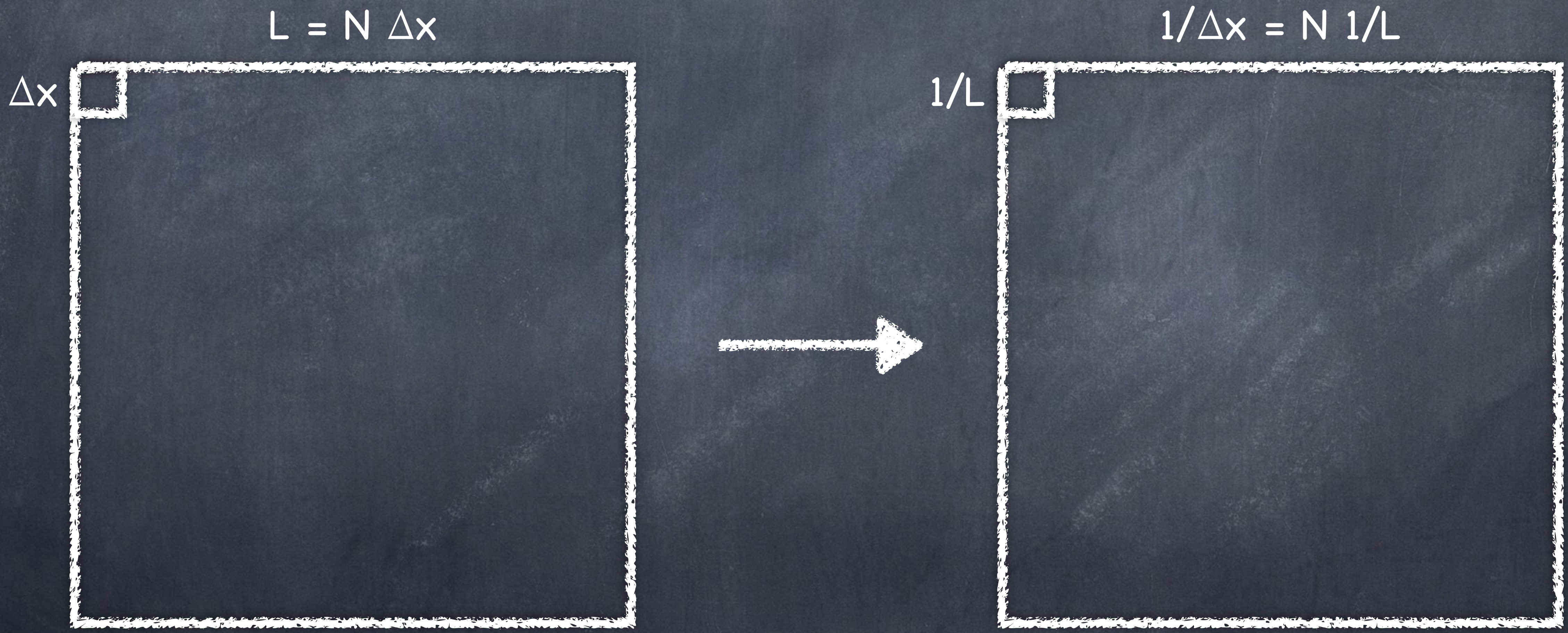


Energy cascade:

wind shear => turbulent energy injected into the system via a large eddy ($\mathcal{L}_0$) which splits into smaller and smaller eddies (ℓ_0), and is finally viscously dissipated.

Interval [ℓ_0, $\mathcal{L}_0$] = inertial range.

(A reminder of discrete Fourier transform (DFT)...)



Images & turbulence — 14

$$\Phi_\varphi(\vec{\nu}) = 0.0228\ r_0^{-\frac{5}{3}} \left(\nu^2 + \frac{1}{\mathcal{L}_0^2}\right)^{-\frac{11}{6}}$$

Which, numerically written, and by considering wavefronts made of ‘dim’ pixels corresponding to ‘L’ meters, becomes:
(re-writing - ``de-dimensionalizing” - the equation with L_0=L_0 L/L and ν=ν L/L…)

```
freq = findgen(dim)
dsp  = .0228*(L/r0)^(5/3.)*L^2*(freq^2+(L/L0)^2)^(-11./6)
```

And which (with the right frequency scale) can be plot with:

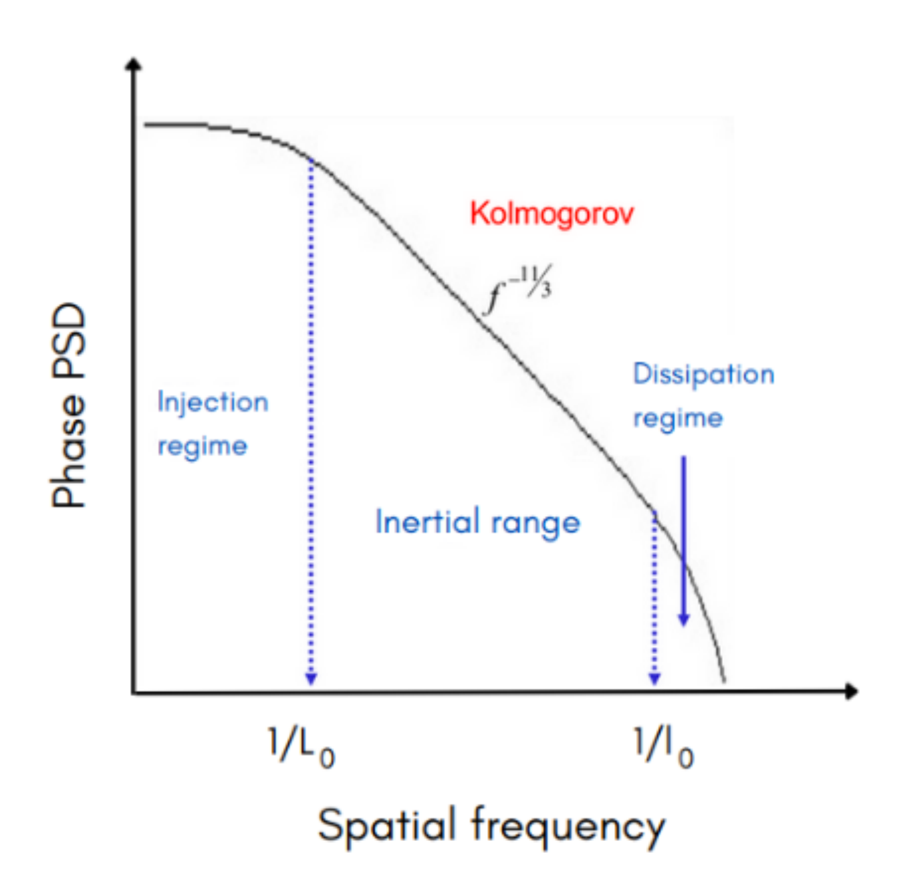
```
plot_oo, 1./L*findgen(dim), dsp, XR=[1/L/1.2,dim*1/L*1.2], /XS
```

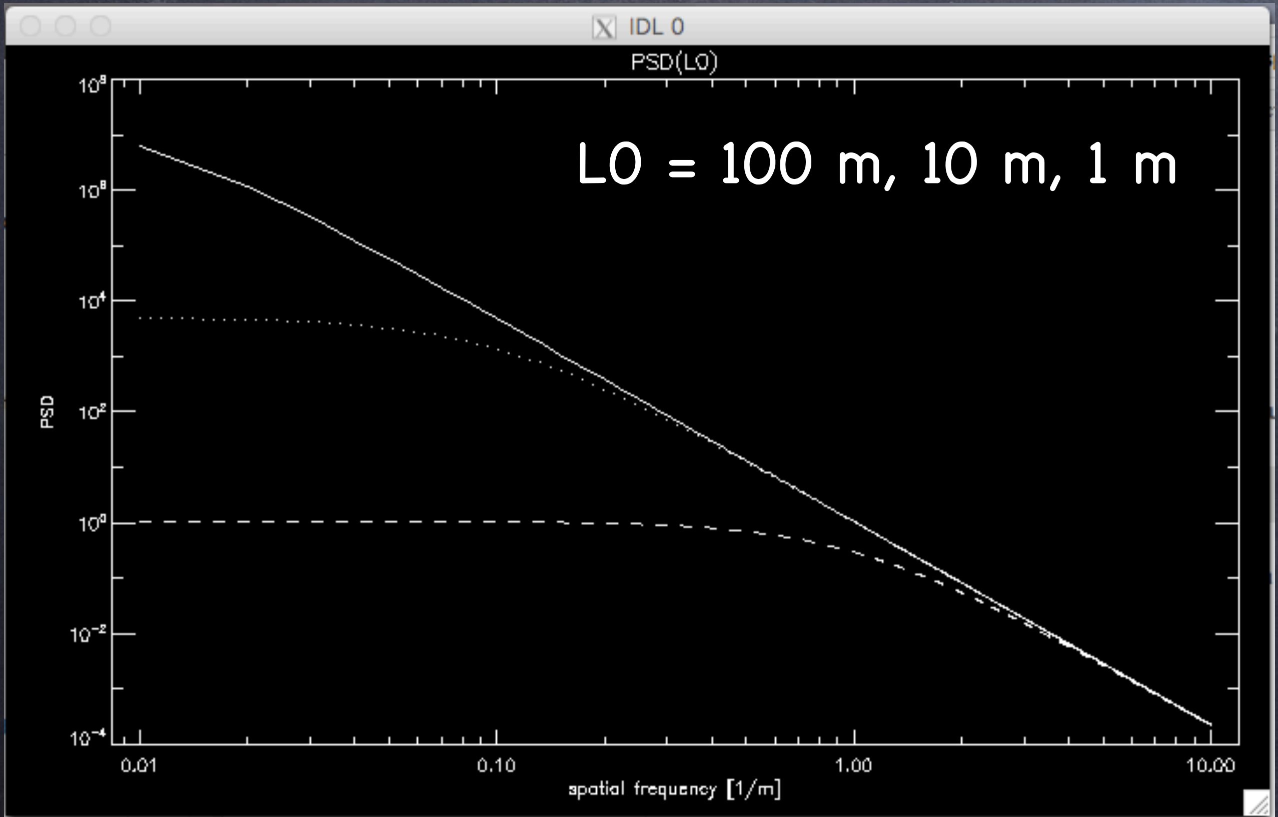
=> make a function that computes PSD(L_0, r_0, dim, L) and plot it for different [r_0, L_0]… [with, for example: dim=1000, L=100., r0=0.1, L0=100.,10.,1.]

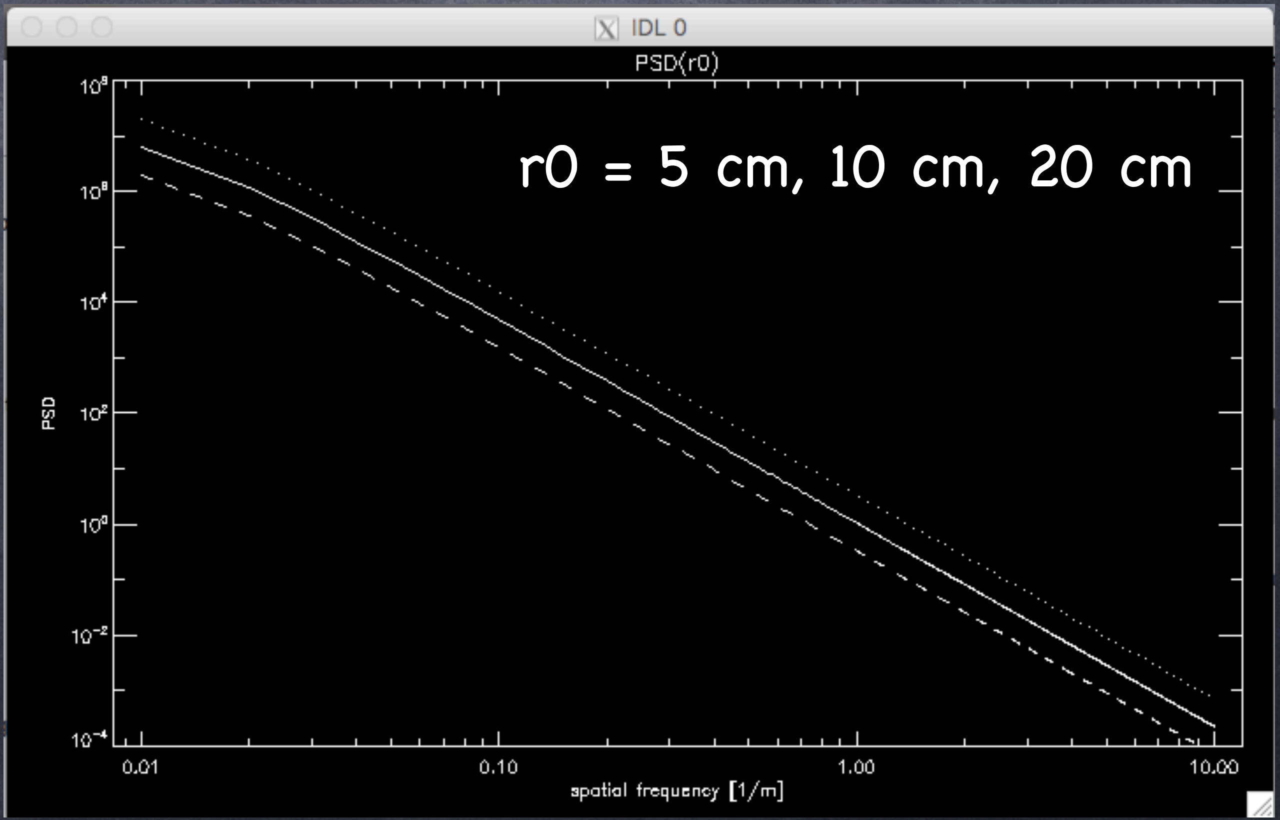
Images & turbulence — 15

```
function dsp_theo, dim, L, r0, L0
;
; dim = array linear dimension [px]
; L   = array physical length [m]
; r0  = phase screen Fried parameter [m]
; L0  = phase screen outerscale [m]
; use: dsp=dsp_theo(dim,L,r0,L0)
; to be plotted afterwards with:
; plot_oo, 1./L*findgen(dim), dsp, XR=[1/L/1.2,dim*1/L*1.2], /XS, $
;          YR=[0.1, 1E7], TIT='PSD(L0)', XTIT='freq. [1/m]', YTIT='PSD'
; oplot  , 1./L*findgen(dim), dsp, LINE=1
; playing, e.g., with L0=100.,10.,1., or r0=.05, .1, .2
;
freq = findgen(dim)
dsp  = .0228*(L/r0)^(5/3.)*L^2*(freq^2+(L/L0)^2)^(-11./6)

return, dsp
end
```







Images & turbulence — 16

-> For next time: read Aime (Sec. 1 & Sec. 2) and Maire (Chap.1)...

Chapitre 1

Introduction

Jérôme Maire, PhD thesis (in French), chap.1

Long Telescopes may cause Objects to appear brighter and larger than short ones can do, but they cannot be so formed as to take away the confusion of the Rays which arises from the Tremors of the Atmosphere.
I. Newton, *1717 Optics, Sec. Ed., Book I, Part I, Prop.VIII*

INSTITUTE OF PHYSICS PUBLISHING EUROPEAN JOURNAL OF PHYSICS

Eur. J. Phys. **22** (2001) 169–184 www.iop.org/Journals/ej PII: S0143-0807(01)14580-0

Teaching astronomical speckle techniques

Claude Aime

UMR 6525 Astrophysique, Faculté des Sciences de l'Université de Nice Sophia-Antipolis, Parc Valrose-06108, Nice Cedex 2, France

Received 7 June 2000, in final form 6 December 2000

Abstract
This paper gives an introduction to speckle techniques developed for high angular-resolution imagery in astronomy. The presentation is focussed on fundamental aspects of the techniques of Labeyrie and Weigelt. The formalism used is that of Fourier optics and statistical optics, and corresponds to graduate level. Several new approaches of known results are presented. An operator formalism is used to identify similar regions of the bispectrum. The relationship between the bispectrum and the phase closure technique is presented in an original geometrical way. Effects of photodetection are treated using simple Poisson statistics. Realistic simulations of astronomical speckle patterns illustrate the presentation.

also have a look here [A. Tokovinin tutorial on atm'c turbulence]:

https://www.ctio.noirlab.edu/~atokovin/tutorial/part1/turb.html

Images & turbulence — 17

-> Perturbed wavefront generation

The well-known FFT method allows us to generate phase screens $\varphi(\vec{r})$, where $\vec{r}$ is the two-dimensional position within the phase screen, assuming usually either a Kolmogorov or a von Karman spectrum $\Phi_\varphi(\vec{\nu})$, where $\vec{\nu}$ is the two-dimensional spatial frequency, from which is computed the modulus of $\tilde{\varphi}(\vec{\nu})$, the Fourier transform of $\varphi(\vec{r})$. Assuming the near-field approximation and small phase perturbation [3], the von Karman/Kolmogorov spectrum is given by

$$\Phi_\varphi(\vec{\nu}) = 0.0229 r_0^{-\frac{5}{3}} \left(\nu^2 + \frac{1}{\mathcal{L}_0^2}\right)^{-\frac{11}{6}}, \qquad (1)$$

where r_0 is the Fried parameter and $\mathcal{L}_0$ is the wavefront outer scale (infinite for the Kolmogorov model). Within the framework of the classical FFT-based technique, a turbulent phase screen $\varphi_L(\vec{r})$ of physical length L is obtained by taking the inverse FFT of $\tilde{\varphi}_L(\vec{\nu})$, the modulus of which is obtained from Eq. (1) by applying the definition of the power spectrum, which is

$$\Phi_\varphi(\vec{\nu}) = \lim_{L\to\infty}\left(\frac{\langle|\tilde{\varphi}_L(\nu)|^2\rangle}{L^2}\right)$$

$$\Rightarrow |\tilde{\varphi}_L(\nu)| \simeq L r_0^{-\frac{5}{6}} \sqrt{0.0228}\left(\nu^2 + \frac{1}{\mathcal{L}_0^2}\right)^{-\frac{11}{12}}, \qquad (2)$$

and which phase is random and uniformly distributed.

(From Carbillet & Riccardi, sec. 2: read it as well…)

(the same manipulation as before is applied here in order to obtain the numerical formulation here below.)

The obtained phase screen is thus numerically written

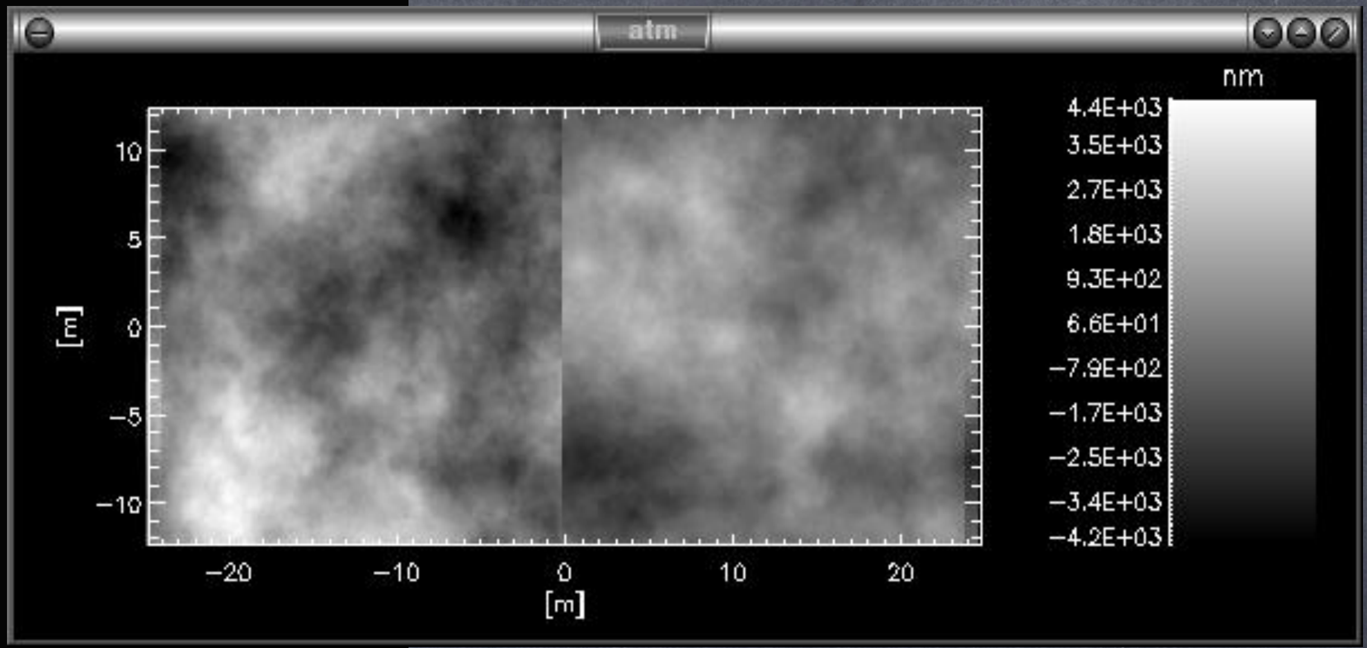
$$\varphi_L(i,j) = \sqrt{2}\sqrt{0.0228}\left(\frac{L}{r_0}\right)^{\frac{5}{6}} \left\{ \mathrm{FFT}^{-1}\left[\left(k^2 + l^2 + \left(\frac{L}{\mathcal{L}_0}\right)^2\right)^{-\frac{11}{12}} \exp\{\imath\theta(k,l)\}\right]\right\}, \qquad (3)$$

where i and j are the indices in the direct space, k and l are the indices in the FFT space, {} stands for either *real part of* or *imaginary part of*, $\imath$ is the imaginary unit, and θ is the random uniformly distributed phase (between $-\pi$ and π). The factor $\sqrt{2}$ comes from the fact that here we use both the real and imaginary parts of the original complex generated FFT phase screens, which are independent one from the other [4]. This kind of phase screen suffers, however, from the lack of spatial frequencies lower than the inverse of the necessarily finite length L of the simulated array.

Images & turbulence — 18

```
function wfgeneration, dim, length, L0, r0, lambda, SEED=seed
;
; wave-front (wf) generation following von Karman model
; (infinite L0 -Kolmogorov model- not allowed here).
;
; dim    = wf linear dimension [px],
; length = wf physical length [m],
; L0     = wf outer-scale [m],
; seed   = random generation seed (OPTIONAL),
; r0     = Fried parameter at wavelength 'lambda' [m],
; lambda = wavelength at which r0 is defined.
;
; Marcel Carbillet [marcel.carbillet@unice.fr],
; lab. Lagrange (UCA, OCA, CNRS), Feb. 2013.
;
; Last modification: March 2025.
;
theta = (randomu(seed,dim,dim)-.5) * 2*!PI    ; rnd uniformly distributed phase
                                              ; (=argument), between -PI and +PI,
                                              ; of the complex number describing
                                              ; the FFT of the phase screen phi
freq  = dist(dim)                             ; spatial frequency array
modul = sqrt(2)*sqrt(.0228)*(length/r0)^(5/6.)*(freq^2+(length/L0)^2)^(-11/12.)
                                              ; von Kármán model
phi   = fft(modul*exp(complex(0,1)*theta), /INVERSE)
                                              ; compute phase screen phi
wf    = phi*lambda/(2*!PI)                    ; phase screen phi => wavefront wf
wf   -= mean(wf)                              ; force mean to zero

return, wf                                    ; deliver 2 independent wf:
                                              ; float(wf) & imaginary(wf)
end
```



<u>wf generation:</u>

generate a cube of statistically independent wf (typically 100)...

=> compute mean *rms* for different [r_0 , L_0]

Images & turbulence — 19

```
function wfcube2, dim, length, L0, r0, lambda, n_wf, filewf

;+
; example of use:
; dim       = 128L         ; [px] wf dimension
; length    = 2.           ; [m] wf physical dimension
; L0        = 27.          ; [m] outerscale of turbulence
; r0        = .1           ; [m] Fried parameter
; lambda    = 500E-9       ; [m] r0 wavelength
; n_wf      = 100L         ; nb of generated wf
; filewf    = 'cube.sav'   ; cube of wf filename
;
; print, wfcube2(128L, 2., 27., .1, 500E-9, 100L, 'wf_r0=10cm_L0=10m.sav')*1E9
; -> compute the cube of wf, save it, and print the rms value in nm
;
; sub-routines needed:
; wfgeneration.pro, compute_rms.pro
;
; Marcel Carbillet [marcel.carbillet@unice.fr],
; lab. Lagrange (UCA, OCA, CNRS), Feb. 2018.
; Last modification: 11th March 2024
;-

; preliminary
cube = fltarr(dim,dim,n_wf) ; initialize cube of wf

; compute and save cube of wf
for i=0, n_wf/2-1 do begin  ; generate wf
    wf = wfgeneration(dim, length, L0, r0, lambda, SEED=seed)
    cube[*,*,2*i]   = float(wf)
    cube[*,*,2*i+1] = imaginary(wf)
endfor
save, cube, FILE=filewf     ; save cube of wf to disk

; compute mean rms
rms = compute_rms(cube)     ; compute rms

return, rms                 ; return back
end
```

```
function compute_rms, cube
; cube: cube of wavefronts (square wf, no pupil!)

n_wf = (size(cube))[3]
rms  = fltarr(n_wf)

for i=0,n_wf-1 do begin
   toto   = moment(cube[*,*,i], SDEV=dummy)
   rms[i] = dummy
endfor

rms_moy = mean(rms)

return, rms_moy
end
```